



# A Line-Based Data Dissemination protocol for Wireless Sensor Networks with Mobile Sink

Elyes Ben Hamida, Guillaume Chelius

## ► To cite this version:

Elyes Ben Hamida, Guillaume Chelius. A Line-Based Data Dissemination protocol for Wireless Sensor Networks with Mobile Sink. IEEE International Conference on Communications (ICC 2008), IEEE, May 2008, Beijing, China. pp.2201 - 2205, 10.1109/ICC.2008.420 . inria-00384139

**HAL Id: inria-00384139**

**<https://inria.hal.science/inria-00384139>**

Submitted on 14 May 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Line-Based Data Dissemination protocol for Wireless Sensor Networks with Mobile Sink

Elyes Ben Hamida  
ARES INRIA / CITI  
INSA Lyon, F-69621, France  
Email: elyes.ben-hamida@insa-lyon.fr

Guillaume Chelius  
ARES INRIA / CITI  
INSA Lyon, F-69621, France  
Email: guillaume.chelius@inria.fr

**Abstract**—In wireless sensor networks, data dissemination is generally performed from the sensor nodes towards a static sink. In this paper, we address the particular case where the sink is mobile, according to an unpredictable mobility pattern. First, we study existing approaches. As an alternative, we present the Line-Based Data Dissemination (LBDD) protocol. Next, we analytically evaluate the communication cost of this protocol and we compare it to other approaches. Finally, realistic simulations are performed and results show that LBDD outperforms previous approaches and presents the best tradeoff among the evaluated protocols.

## I. INTRODUCTION

A Wireless Sensor Network (WSN) is a multi-hop wireless network composed of sensor nodes. A sensor node is generally a constrained device with relatively small memory, restricted computation capability, short range wireless transmitter-receiver, and limited built-in battery. Furthermore, sensor networks usually operate on an N-to-1 communication paradigm, where sensors collect environmental data and forward them towards a static base station or *sink*. This procedure is called *data dissemination*. In many situations, a static sink may be unfeasible because of deployment or security constraints. Sink mobility may also improve the lifetime of a WSN by avoiding excessive transmission overhead at nodes that are close to the location that would be occupied by a static sink [1]. The sink mobility assumption may be useful for many applications such as target tracking, emergency preparedness, and habitat monitoring. In such a context, the difficulty for sensor nodes is to efficiently track the sink and report to it. As flat architectures and flooding-based protocols do not scale, overlaying a virtual infrastructure over the physical network has often been investigated as an interesting strategy for efficient data dissemination. This virtual infrastructure acts as a rendez-vous area for queries and data reports.

Several protocols have been proposed to implement a scalable and energy-efficient data dissemination architecture for WSNs. GHT (Geographic Hash Table) [2] has introduced the concept of data-centric storage (DCS). GHT hashes a *sensed-data-type* into geographic coordinates and stores the corresponding data at the sensor node (*a.k.a. home-node*) the closest to these coordinates. Two-Tier Data Dissemination (TTDD) [3] provides a scalable and efficient data delivery to multiple mobile sinks. Each source pro-actively builds a

grid structure by dividing the sensor field into cells with dissemination nodes located at the crossing points of the grid. Queries and data are then transmitted along the grid. Railroad [4] builds and exploits a virtual infrastructure, called a rail. This rail is placed in the middle area of the sensor field so that each node can easily access it. When a source detects a new event, the data remains locally stored and the corresponding meta-data is sent to the rail. This infrastructure is then used by the mobile sink to retrieve meta-data, with the queries traveling around the rail. The Column-Row Location Service (XYLS) [5] is a proactive location service which can be used to disseminate data reports. In XYLS, source nodes disseminate and replicate the data reports in the north and south directions from their current location, such that sink queries can intersect it subsequently.

In this paper, we present the Line-Based Data Dissemination (LBDD) protocol and compare it to existing approaches for data dissemination in WSNs with mobile sinks. LBDD relies on the concept of a rendez-vous region to decouple the operation of *data dissemination* from the one of *data collection*. We show that LBDD outperforms other approaches in *event-driven* and *query-based* scenarios. In general, we also show how the choice of an efficient dissemination structure may impact on dissemination protocol performances.

The remainder of this paper is organized as follows. Section II describes the LBDD protocol. In Section III, the analytical communication costs of LBDD and existing approaches are evaluated and compared. Simulation results are analyzed in Section IV. Section V finally concludes this article.

## II. THE LINE-BASED DATA DISSEMINATION PROTOCOL

The Line-Based Data Dissemination (LBDD) protocol defines a vertical virtual *line* of width  $w$  which divides the sensor field into two parts, as shown in Fig. 1. This line is also divided into groups of size  $g$ . These two parameters are used to address the hotspot problem and the scalability issue. The *line* is placed in the center of the sensor field so that each node can easily access it. Nodes within the boundaries of this wide line are called *inline-nodes*, while the other nodes are referred to as *ordinary nodes*. This *line* acts as a *rendez-vous* region for data storage and lookup.

As in [2]–[4], the LBDD design assumes that each node knows its geographic location as well as the network ge-

ographic boundaries. This is not an utopian assumption as in most sensor applications (*e.g.*, target tracking, military surveillance, *etc.*) the data are strongly correlated to the geographic location. A node position can be obtained through the use of GPS or some virtual coordinate system. Eligibility of nodes as *inline-nodes* is then easily performed based on this geographic information. Thus, the overhead for building the virtual infrastructure is avoided. Finally, we suppose that there are multiple sinks moving randomly in the sensor field.

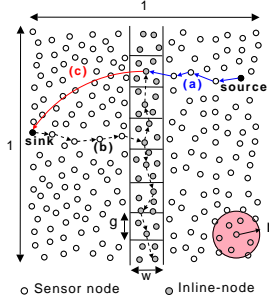


Fig. 1. The LBDD virtual infrastructure.

The operation of LBDD is composed of two main steps: (i) **Dissemination**: when an ordinary sensor node generates some new data, it forwards the data to the nearest inline-node (see Fig. 1(a)); and (ii) **Collection**: in order to retrieve a specific data, a sink sends a *query* towards the line in a perpendicular fashion. The first inline-node which receives this query propagates it in both directions along the *line* until it reaches the inline-node storing the data (see Fig. 1(b)). The data is then sent directly to the sink (see Fig. 1(c)). To avoid the transfer of duplicated data, we suppose that a sink receives a response to its query only if the *inline-node* owns a new data.

To ease the data lookup process, two data-storage schemes are possible: the data can be either stored in all nodes of a group or just in their group-leader. The first scheme needs a fine-tuning of  $w$  and  $g$  to prevent an increase of the congestion under high traffic load conditions, while the second one requires a periodic group-leader election and a replication mechanism. Finally, the sink mobility is supported using the *progressive-footprint-chaining* strategy [6].

Several optimizations are possible to increase the efficiency of LBDD. First, in scenarios where the number of queries is higher than the number of data reports, replicating the data on the entire virtual line could decrease the cost of data-lookup. In the rest of this paper we will refer to LBDD\_R as the basic LBDD protocol with data replication. Second, in *target-tracking* scenarios, a mobile target can evolve and remain in a given area. In that case, once the sink receives a data report from a given *inline-node*, the next sink query's can be sent directly to this inline-node instead of covering the entire virtual infrastructure. This allows to decrease the overhead and to minimize the energy consumption inside the virtual infrastructure.

### III. PERFORMANCE ANALYSIS

This section provides an average-case communication cost analysis. This communication cost represents the total amount of messages generated in the network during the data dissemination and lookup process. We first present the models and assumptions, then we evaluate the average distances and the communication cost of LBDD and we compare them to GHT [2], RailRoad [4], TTDD [3] and XYLS [5].

#### A. Models and assumptions

We consider a network with a large number of nodes being uniformly and independently distributed over a  $1 \times 1$  area. We define  $H(l)$  as the number of hops on a path between two arbitrary nodes  $x$  and  $y$  such that  $|x, y| = l$ . According to [7], given a geographic routing protocol, we have  $H(l) = \zeta \frac{l}{r}$  with  $r$  the communication range and  $\zeta \geq 1$  a scaling factor which depends on the density of nodes. For numerical applications, we will assume that:  $\zeta = 1$ .

The communication cost of data dissemination and data collection is proportional to the distance between the different communicating entities. Then, to compare LBDD to other protocols, it is important to evaluate the distances between the protocol entities. In the rest of the paper we define the following metrics: (i)  $\mathbb{D}_{src,rdv}$  is the distance between the source-node and the rendez-vous area; (ii)  $\mathbb{D}_{sink,rdv}$  is the distance between the sink and the rendez-vous area; and (iii)  $\mathbb{D}_{rdv,sink}$  is the distance between the rendez-vous area and the sink. We assume that  $\mathbb{D}_{sink,rdv}$  and  $\mathbb{D}_{rdv,sink}$  may be different as a query and the data transfer do not necessarily use the same path. We consider four types of messages: event notification, query, data, and control messages, whose sizes are  $p_e$ ,  $p_q$ ,  $p_d$ , and  $p_c$ , respectively. We consider  $m$  sinks moving randomly in the sensor field as well as  $n$  sources. Each sink generates a number of queries equal to  $\bar{q}$  and each source generates a number of events equal to  $\bar{e}$ . Thus, the total number of queries and events can be written as  $m\bar{q}$  and  $n\bar{e}$ . Finally, we disregard the LBDD line's width for the theoretical evaluation.

#### B. Average distances evaluation

**LBDD**: In the average-case, a source node is located at a distance  $\mathbb{D}_{src,rdv} = 0.25$  from the virtual line for data dissemination. To compute the average distance from the sink to the rendez-vous point (*i.e.*, the *inline-node* which stores the requested data), we evaluate the average distance between the sink and the virtual line, which is equal to 0.25, plus the average distance covered by the query inside the line during the data lookup process.

Let  $Y_1$  and  $Y_2$  the Y-coordinates of two randomly chosen points in the unit square. We consider the following random variable:  $D = |Y_1 - Y_2|$  with a PDF  $f_D(d) = 2(1 - d)$  for  $0 < d < 1$  and 0 otherwise. The average distance covered by a query inside the virtual line is equal to  $\mathbb{E}[D] + \frac{1 - \mathbb{E}[D]}{2}$ . The average distance between the sink and the inline-node is finally equal to:  $\mathbb{D}_{sink,rdv} = 0.25 + \frac{1}{3} + \frac{1}{3} \approx 0.9166$ .

The average distance from the inline-node to the sink for the data transfer is equal to the average distance between

two randomly chosen points in a unit square and the central line, respectively. This distance can be expressed as follows:  $D = \sqrt{|X_1 - 0.5|^2 + |Y_1 - Y_2|^2}$ , with  $X_1$ ,  $Y_1$  and  $Y_2$  three independent and uniform random variables. The random variable  $D$  is defined by the following PDF [8]:

$$f_D(d) = \begin{cases} 2d(\pi - 2d) & 0 < d \leq \sqrt{0.25} \\ 2d(\arccos(1 - \frac{1}{2d^2}) - 1) & \sqrt{0.25} < d \leq 1 \\ 2d(\arccos(1 - \frac{1}{2d^2}) - \arccos(\frac{2}{d^2} - 1)) & 1 < d \leq \sqrt{1.25} \\ 0 & \text{otherwise} \end{cases}$$

The average distance is evaluated numerically using a Riemann sum and is computed as  $\mathbb{D}_{rdv,sink} \approx 0.45$ .

**Comparison with other protocols:** The average distances of the other protocols are computed in a similar way. For further details please refer to [8]. Results are summarized on Tab. I. From these results we can make three observations. First, we notice that a random rendez-vous node (*i.e.*, GHT) presents a lower distance for data lookup ( $\mathbb{D}_{sink,rdv}$ ) than the other infrastructures (except replication-based solutions). As the virtual infrastructure is limited to a single node, there is no need to search for the requested data and the cost of data lookup is then avoided. However, this characteristic may induce a hotspot problem and causes congestion (*i.e.*, lost of packets) as all queries and data reports are concentrated on a single node.

Protocols	$\mathbb{D}_{src,rdv}$	$\mathbb{D}_{sink,rdv}$	$\mathbb{D}_{rdv,sink}$	$\mathbb{P}_{stretch}$
GHT [2]	0.52	0.52	0.52	1.5
TTDD [3]	0.0	0.88	0.88	1.69
Railroad [4]	0.09	1.46	0.51	1.98
XYLS [5]	1.0	0.66	0.33	0.96
LBDD	0.25	0.91	0.45	1.23
LBDD_R	1.25	0.25	0.25	1.04

TABLE I  
AVERAGE DISTANCES EVALUATION FOR EXISTING APPROACHES.

Second, the use of a large virtual infrastructure like LBDD, TTDD or Railroad decreases the cost of dissemination ( $\mathbb{D}_{src,rdv}$ ) compared to the random rendez-vous area. Indeed, as the size of the rendez-vous area increases, source nodes get usually closer to this infrastructure. In addition, using a large infrastructure allows to distribute the communication load through the nodes belonging to the rendez-vous area. However, the use of large infrastructures for data dissemination induces a higher cost for the lookup ( $\mathbb{D}_{sink,rdv}$ ) as the data need to be searched. Third, replication-based solutions, like LBDD\_R and XYLS, allow to minimize the cost of data lookup and collection ( $\mathbb{D}_{rdv,sink}$ ). However, the replication induces an increase in the dissemination cost, while minimizing the average path-stretch.

From this analysis a tradeoff emerges between the virtual infrastructure size and the data dissemination/lookup path distances. Large virtual infrastructures are more suitable to applications inducing a large number of data reports compared to the number of queries (*e.g.*, *periodic sensing* or *event driven* applications), while small infrastructures surpass the latter in

scenarios with a large number of queries compared to the number of data reports (*e.g.*, *query based* applications). This fact is further analyzed in the following Section.

### C. Communication cost

The communication cost represents the total amount of messages generated in the network during the data dissemination and lookup processes. It is defined as:  $C_{\text{protocol}} = C_{\text{DD}} + C_{\text{DL}} + C_{\text{DT}}$ , where  $C_{\text{DD}}$ ,  $C_{\text{DL}}$ ,  $C_{\text{DT}}$  are the costs of data dissemination, data lookup, and data transfer, respectively. The subscript "protocol" refers here to one of the solutions listed in Table I. To better investigate the tradeoff analyzed in previous section, we consider two *application-scenarios* for evaluating the communication costs. In *scenario 1*, we consider an *event-driven* application with a fixed average number of query per sink ( $\bar{q} = 50$ ) while varying the average number of data reports per source. In *scenario 2*, we consider a *query-based* application with a fixed average number of data report per source ( $\bar{e} = 50$ ) while varying the average number of queries per sink. For both scenarios we consider 10,000 sensor nodes deployed on a square sensor field of size  $1 \times 1$ . The sensor coverage area radius is  $r = 0.1$  and we suppose that the size of a TTDD cell,  $\alpha$ , is 0.3.

According to Tab. I, the total average communication cost of LBDD is defined as:  $C_{\text{LBDD}} = n\bar{e}p_d H(0.25) + m\bar{q}p_q H(0.91) + n\bar{e}p_d H(0.45)$ , where  $n\bar{e}p_d H(0.25)$  is the cost of the data dissemination,  $m\bar{q}p_q H(0.91)$  is the cost of the data lookup, and  $n\bar{e}p_d H(0.45)$  is the cost of data transfer. The communication costs of GHT, XYLS, Railroad, and TTDD are computed in a similar way (for further details, refer to [8]).

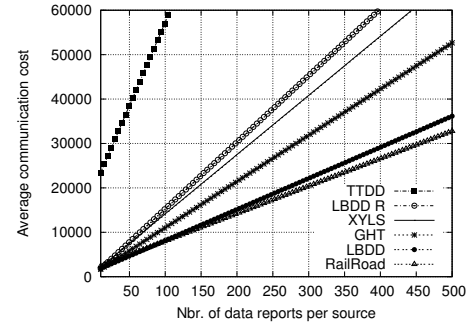


Fig. 2. Average communication cost for scenario 1 ( $\bar{q} = 50$ ,  $m = 5$  sinks,  $n = 10$  sources,  $p_e = p_c = p_q$ ,  $p_d = 2 \times p_q$ ).

**Scenario 1.** The average communication costs for *scenario 1* are shown in Fig. 2. We notice that TTDD presents a rather high communication cost stemming from its need to build grids and its routing strategy along the grid. Moreover, we observe that the replication mechanism of LBDD\_R and XYLS engenders a high communication cost, as the number of data reports increases. On the other side, protocols implementing a large virtual infrastructure like Railroad and LBDD are more suitable to scenarios with a high number of data reports. However, we notice that for a high number of data reports per source node ( $\geq 200$  data reports per source),

RailRoad performs slightly better than the others protocols. As RailRoad implements a large infrastructure (*i.e.*, a square of width  $l = 0.7$ ), sensor-nodes are closer to the rendez-vous area and the cost of data dissemination is reduced.

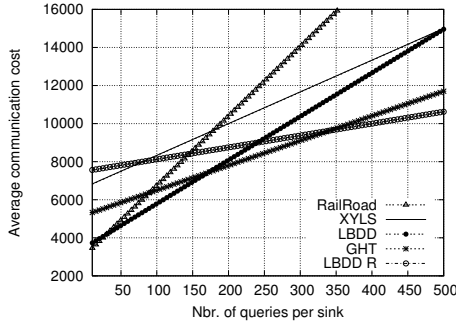


Fig. 3. Average communication cost for scenario 2 ( $\bar{e} = 50$ ,  $m = 5$  sinks,  $n = 10$  sources,  $p_e = p_c = p_q$ ,  $p_d = 2 \times p_q$ ).

**Scenario 2.** The average communication costs for *scenario 2* are shown in Fig. 3. Note that TTDD presents a rather high communication cost and is not represented on this figure. RailRoad and XYLS also present a high communication cost as the number of queries increase. This cost is due to the increase of the data lookup cost. In addition, we notice that for  $\bar{q} \leq 170$ , the average communication cost of LBDD is lower than the ones of the other approaches. Then, for  $\bar{q} \in [170 \dots 340]$ , GHT presents a lower average communication cost. And finally for a high number of query per sink ( $\bar{q} \geq 340$ ), the replication mechanism of LBDD\_R allows to decrease the communication cost compared to the other schemes. The reason is that the more the number of queries increases, the more the cost of data lookup becomes crucial. Replication of data reports on the virtual infrastructure allows in that case to minimize the total cost. For example, to find the threshold from which the replication mechanism of LBDD becomes beneficial, we have to satisfy this condition :

$$\frac{C_{LBDD}}{C_{LBDD\_R}} > 0 \Leftrightarrow \bar{q} > 1.21 \frac{n p_d}{m p_q} \bar{e} \quad (1)$$

This relation between the number of queries and data reports (Eq. 1) can be easily used by LBDD or other approaches to decide when a replication mechanism has to be used or not. In that case, each group-leader (*i.e.*, one of the inline-nodes) evaluates for each data type the number of received queries' and data reports during a given period of time, and depending on Eq. 1 can adapt its behavior *on-the-fly* to cope with the application needs. Such an adaptive protocol could decrease the overhead and congestion inside the virtual infrastructure.

#### IV. REALISTIC SIMULATIONS

For the purpose of performance evaluation under more realistic network factors (*e.g.*, collisions, interferences, radio propagation, *etc.*) we have carried out simulations. The objective is to investigate tradeoffs involving congestion as well as energy consumption. A comparative study is performed between GHT, XYLS and LBDD with a varying number of

source-sink pairs. In the simulations, we used the basic version of LBDD without the replication mechanism.

##### A. Simulation environment

In order to guarantee a fair comparison between LBDD, XYLS, and GHT, we have implemented the three protocols in the WSNNet<sup>1</sup> simulator. Table II presents the simulation parameters.

Parameters	Values
MAC protocol	IEEE 802.11 DCF
Radio	BPSK, TX $\approx -20dBm$ , sensitivity = $-70dBm$
Propagation	Free-space with a pathloss $\beta = 3$
Interference model	SINR/Shotnoise model (Full interference model)
Routing protocol	greedy geographic routing
Sensor field size	$500 \times 500m^2$
Node density	1400 nodes
Scenario 1	1 query per 1s and 1 data report per 0.5s
Scenario 2	1 query per 0.5s and 1 data report per 1s

TABLE II  
SIMULATION PARAMETERS.

We consider two scenarios : (i) *scenario 1*: with a sink data-rate of 1 query per second and a source data-rate of 2 data-packet per second, and (ii) *scenario 2*: with a sink data-rate of 2 query per second and a source data-rate of 1 data-packet per second. Source nodes are randomly deployed into the sensor field and a *Linear mobility model* is used for the sinks with an average speed of  $10m/s$ . Given that GHT was not designed to support sink mobility, we also use the *progressive footprint chaining*. In addition, each GHT source-sink pair chooses a different randomly placed *home-node* at each run. This *home-node* is then used by all sensor nodes as a rendez-vous area for queries and data reports. The LBDD virtual infrastructure's parameters,  $g$  and  $w$ , are set to  $50m$  and  $50m$ , respectively. All the simulation results are averaged over 10 different runs.

We use four metrics for the performance evaluation of the three protocols: (i) *Energy consumption*: defined as the global energy consumption during a simulation; (ii) *Data delivery ratio*: defined as the ratio between the total number of data packets received by the sink and the total number of data generated by its corresponding source; and (iii) *Delay*: defined as the total time elapsed between the data generation by a source and its reception by a sink.

##### B. Simulation results

Fig. 4(c-f) shows the average energy consumption which impacts strongly the network lifetime. For the three protocols, we notice that the energy cost is linear in the number of pairs as the source-sinks pairs are independent. We observe that XYLS presents a rather high communication cost in both scenarios. This is a direct consequence of the data replication along the vertical line. Next, we notice that LBDD presents a lower energy consumption than GHT on *scenario 1* (see Fig. 4(c)). Given that all source-sink pairs in GHT have the same *home-node*, the average path length between the sink and the source via this *home-node* is higher than the average

<sup>1</sup>The WSNNet simulator is available at: <http://wsnet.gforge.inria.fr/>

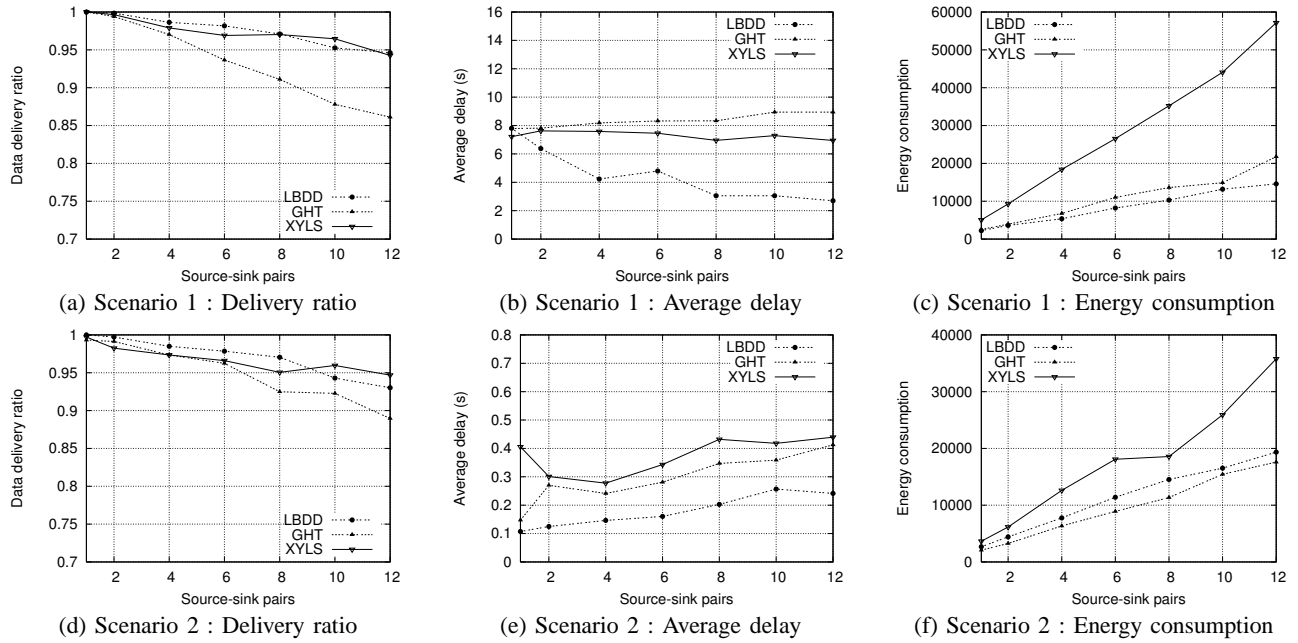


Fig. 4. Simulation Results.

distance in LBDD. This result confirms the analysis made for the average case and presented in section III. However, on *scenario 2* GHT performs better than LBDD (see Fig. 4(f)). As in this scenario the number of queries is twice the number of data reports, the cost of data lookup inside the virtual infrastructure increases the total communication cost of LBDD compared to GHT.

Fig. 4 (a-d) shows the average data delivery ratio. We notice that the success rate slightly decreases as the number of source-sink pairs increases because of congestion in the network. We notice on both scenarios that GHT presents a lower delivery ratio. Given that all data reports and queries converge towards the same single *home-node*, the congestion as well as the collisions increase. On the other side, LBDD and XYLS presents a higher delivery ratio stemming from the use of a virtual infrastructure which allows to better distribute the load among the nodes inside the rendez-vous area. However, in *scenario 2*, we observe that the delivery ratio of LBDD decreases slightly compared to *scenario 1*. But the delivery ratios of LBDD and XYLS remain close in both scenarios. Fig. 4 (b-e) shows the average delay. We notice on both scenarios that LBDD presents a lower delay compared to the other protocols.

## V. CONCLUSION

In this paper we analyzed and compared various approaches for data dissemination in wireless sensor networks with mobile sinks. We also proposed a new virtual infrastructure called LBDD allowing an efficient data dissemination. LBDD relies on the concept of a rendezvous region to decouple the operation of *data dissemination* from the one of *data collection*. Through analytical analysis and realistic simulations, we showed that LBDD outperforms other approaches both in

*event-driven* and *query-based* scenarios, and that it presents the best tradeoff among the evaluated protocols.

In the future, we plan to better investigate the problem of data persistence against node failure and malicious nodes inside the virtual infrastructure. We also plan to investigate a near optimal structure for the data dissemination and lookup problem.

## REFERENCES

- [1] W. Wang, V. Srinivasan, and K.-C. Chua, "Using mobile relays to prolong the lifetime of wireless sensor networks," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2005, pp. 270–283.
- [2] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu, "Ght: A geographic hash table for data-centric storage in sensor networks," in *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'02)*, September 2002.
- [3] H. Luo, F. Ye, J. Cheng, S. Lu, and L. Zhang, "TTDD: Two-tier data dissemination in large-scale wireless sensor networks," *ACM Journal of Mobile Networks and Applications (MONET)*, Special Issue on ACM MOBICOM (2003), 2003.
- [4] J. H. Shin, J. Kim, K. Park, and D. Park, "Railroad: virtual infrastructure for data dissemination in wireless sensor networks," *Proceedings of the 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN'05)*, pp. 168–174, Oct. 2005.
- [5] Y. H. S. Das, H. Pucha, "Performance comparison of scalable location services for geographic ad hoc routing," in *Proceedings of IEEE INFOCOM 2005*, Miami, FL, March 13-17 2005, pp. 1228–1239.
- [6] C.-C. Shen, C. Srisathapornphat, and C. Jaikaeo, "Sensor Information Networking Architecture and Applications," *IEEE Personal Communication Magazine*, vol. 8, no. 4, pp. 52–59, August 2001.
- [7] S. De, A. Caruso, T. Chaira, and S. Chessa, "Bounds on hop distance in greedy routing approach in wireless ad hoc networks," *International Journal of Wireless and Mobile Computing*, vol. 1, no. 2, pp. 131–140, 2006.
- [8] E. B. Hamida and G. Chelius, "Analytical evaluation of virtual infrastructures for data dissemination in wireless sensor networks with mobile sink," in *SANET '07: Proceedings of the First ACM workshop on Sensor and actor networks*, Montreal, Quebec, Canada, 2007, pp. 3–10.